DOCUMENT RESUME

ED 274 184 FL 016 023

AUTHOR Parkhurst, Christine
TITLE Using C.A.L.L. to Teach Composition.
PUB DATE Mar 84
NOTE 7p.; In: On TESOL '84. A Brave New World for TESOL. Selected Papers from the Annual Convention of the Teachers of English to Speakers of Other Languages (18th, Houston, Texas, March 6-11, 1984); see FL 016 005.
PUB TYPE Reports - Descriptive (141) -- Speeches/Conference Papers (150)

EDRS PRICE MF01/PC01 Plus Postage.
DESCRIPTORS *Artificial Intelligence; Cloze Procedure; *Computer Assisted Instruction; *Computer Software; Editing; *English (Second Language); Higher Education; Revision (Written Composition); Second Language Instruction; Tenses (Grammar); Writing (Composition); *Writing Exercises; *Writing Instruction
IDENTIFIERS Proofreading; University of Massachusetts Boston

ABSTRACT

While artificial intelligence is far from producing a computer program that can understand English, there are computer applications for teaching composition that go beyond the electronic flashcard and other limited applications. A number of programs designed for use in college freshman English as a second language classes based on the artificial intelligence concept of limited domain include software for diagnostic proofreading, a modified cloze exercise correlating grammar with student ability to proofread and make decisions about grammar based on clarity of meaning, a controlled composition exercise on verb tense, composition requiring different past tense forms, and free composition emphasizing topic and commenting on content as well as grammar. These programs meet some of the criteria for an artificially intelligent tutoring system in that they (1) are not limited by the need to input a single correct answer, but respond meaningfully to a wide variety of correct and incorrect answers; (2) gain and use information from users by either asking questions or making their own analysis of input; (3) in some cases respond to input in terms of global paragraph meaning, as a human being would; and (4) can help the student evaluate his own writing in a simulated human fashion. (MSE)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

PERMISSION TO REPRODUCE THIS
MATERIAL HAS BEEN GRANTED BY
J. Frank-McNeil

TO THE EDUCATIONAL RESOURCES
INFORMATION CENTER (ERIC)

USING C.A.L.L. TO TEACH COMPOSITION

ED274184

U.S. DEPARTMENT OF EDUCATION
Office of Educational Research and Improvement
EDUCATIONAL RESOURCES INFORMATION
CENTER (ERIC)

This document has been reproduced as received from the person or organization originating it.

Minor changes have been made to improve reproduction quality.

Points of view or opinions stated in this document do not necessarily represent official OERI position or policy.

Christine Parkhurst
University of Massachusetts/Boston

Four years ago I began writing teaching programs at the University of Massachusetts/Boston. These programs are used as a tutorial adjunct to ESL freshman composition. Like many people, I began by writing programs which review basic grammar. Although they were well received by students, they were typical electronic flashcard exercises. The correlation between improvement on grammar exercises and improved grammar in composition is not self-evident. Since my goal is to help students with composition, I developed five types of composition exercises which are more directly related to composition skills.

Before I describe these programs in detail, I'll discuss the problem all composition programs must face. Then I'll briefly describe solutions other people have found. The central problem is this: Artificial Intelligence is still many years away from the creation of a program which can understand English. (Thompson and Thompson 1975). Is it worthwhile dealing with composition if it's impossible to deal with the meaning of what a student has written?

There are several solutions to this problem. One is to facilitate the revision process by using a word processor such as Word Star or Bank Street Writer. This leaves the student with the responsibility of coming up with a first draft and deciding how and when to revise it. However, revision is so much easier using a word processor that students are encouraged to revise a lot. Another solution is to fake it. Some programs help students to come up with a first draft by asking questions, and offering encouraging comments such as "How interesting," or "Tell me more," no matter what the student has written, rather like comments we make on papers at three A.M. These programs do help students overcome the hurdle of confronting a blank piece of paper. A third solution is to make general stylistic comments based on analysis of grammar and lexicon. This is the approach used by programs such as Bell Laboratories' *The Writer's Workbench*, which is based on Strunk and White's (1972) *The Elements of Style*. If the length of the average sentence is "too long", this type of program may accuse the writer of

FL016023

Christine Parkhurst teaches ESL and Linguistics (Massachusetts College of Pharmacy and University of Massachusetts/Boston). She has written grammar and composition courseware since 1979.

BEST COPY AVAILABLE

being "too wordy". If the ratio of nouns to verbs is too high, the style is labeled unclear. If certain Latinate words are used, the writer is told what shorter word to substitute. Since the program can't understand meaning, these comments may or may not be good guesses. Students who use these programs may focus on mechanics rather than meaning when they write, and become obsessed with sentence length, for example, rather than clarity.

Clearly, using computer programs to comment on composition is problematic. Students relate to computers differently than they do to textbooks or to teachers, and this in itself can help some students with their writing. The difficulties are worth overcoming for this reason.

I am going to describe some of the programs I've written, first in general, then in detail. My programs use a solution to the problem of meaning which is based on the Artificial Intelligence concept of the limited domain. Some programs can understand English fairly well when it relates to a specific, limited topic—a limited domain. Examples are programs which can "understand" baseball statistics or questions about airline ticketing (Woods 1978). My programs use specific paragraphs as their limited domain—they are relatively intelligent about the meaning of one particular paragraph or series of paragraphs. Therefore they can help students learn proofreading skills or evaluate their own writing, using clarity of meaning as one of the criteria. The other idea borrowed from Artificial Intelligence is the type of semantic trickery found in the Eliza program, which stimulates a Rogerian psychoanalyst. It's possible for a program to ask intelligent and relevant questions without understanding the answer. My programs may also ask the questions, and understand and use the information provided in the answer.

The first program type is a diagnostic proofreading program based on a summary written by an ESL class. The student user sees a paragraph with many typical ESL errors, and is invited to read and correct the paragraph. The premise is that an error which a student fails to correct in one way or another is an error the student may also fail to spot and correct when proofreading his own writing. A wide variety of correct answers is anticipated and accepted, and a wide variety of errors is also anticipated and commented on. The program keeps a tally of the types of error, and at the end suggests to the student which grammar areas with which the student has problems. Students who use the programs on a library basis are told which programs to call, and later programs use this information as would a composition teacher who is familiar with a student's weak areas.

The next program type is a modified cloze exercise that attempts to correlate grammar to the student's ability to proofread and make decisions about grammar which are based on clarity of meaning. For example, the student is given a chance to review article use, then given a choice of three paragraphs: easy, medium and difficult. The articles are missing, and the student adds them. The program anticipates all answers, and explains incorrect answers—specifically relating them to the meaning of the paragraph. If you look at the student's answer and the program's comments in the following examples, you'll see that the program is intelligent about the meaning of the paragraph, which describes an unpleasant dinner at a restaurant. It is also intelligent about student input in this limited domain, and is able to give a hint about an incorrect answer, as a human tutor would. The student is learning about article use, and also about proofreading: each sentence must be evaluated.

Student: Maybe there was strike at restaurant.
Computer: Want another try?
Student: No.
Computer: Give up? Here's the answer.
Maybe there was a strike at the restaurant.
'A strike' is correct–
This is the first time this strike was mentioned.
'The restaurant' is correct–
It's the restaurant we've been talking about.
Computer: After hour we left.
Student: After a hour we left.
Computer: Think about how you pronounce 'hour.'
Want another chance now?
Student: Yes
Computer: Good luck!
After hour we left.
Student: After an hour we left.
Computer: Excellent! You get a bonus point!

The third program type is a controlled composition exercise which works on verb tense. Again, grammar is related to meaning. One paragraph must be simple past tense because it happened yesterday. If we change the chronology of the narrative, the tense of the new paragraph changes to past perfect to reflect the change in meaning. The program "understands" the meaning, so it can comment on grammar (for example, "had had" uses "have" as main and auxiliary verb), and also on meaning (for example, a switch to simple present is necessary when the meaning becomes general.) The program then helps the student write and proofread the paragraph, which requires past perfect. Students learn to choose a verb tense which conveys their intended meaning when they write, as the following example of feedback shows:

Computer: I think I failed it.
Student: I had thought I had failed it.
Computer: This one's extra tricky.
'I think I failed it.' is correct.
This didn't happen before lunch–
You think so now!
You always have to think about what you mean and make the tense match the meaning.

The next program type begins to move toward a freer type of composition. This program helps students with a composition requiring different forms of past tense, and helps proofread for correct use of tense. However, students make many different types of improvements when they revise for which they take the initiative. As you can see from this example, students may revise extensively on the sentence level, adding information, clarifying ideas, and using more complex sentence structure as well as correcting grammar. (This is one student's input, not an amalgam.)

First draft: Saturday morning I was studing all day.
Second draft: Saturday morning I was studing all morning.

First draft: I went to Chinise resturan. I had a dinner.
Second draft: I went to chinise resturant and I had a dinner.

First draft: I had some homeworks and I did befor I went to bed.
Second draft: On saturday evening I did my homework befor I went to bed.

The next program type is a free composition program in the sense that the topic is emphasized and the program makes comments on content as well as grammar. What this program attempts to teach is the writing process rather than how to write a biographical essay, article use, etc. To illustrate the use of this program, I have chosen randomly a very weak writer I now have in the second term of Freshman Composition. She was in the Computer Center when I was writing this. Given an in-class composition, she would probably stop with Paragraph A, and look miserable. She is what Krashen would call a monitor underuser—she cannot analyze her own problems (Krashen 1982). In Krashen's model her affective filter is very high when she tries to write. She thinks of herself as a poor writer and becomes so tense she cannot proceed. She also suffers from what I would like to call a *passivity problem*. Poor writers look to their teachers to take charge of the writing process, to decide when and how to revise *for* them. This program offers suggestions and guidance, but it cannot rewrite *for* the student. The student takes responsibility for revision, and this may change the student's attitude. Students become increasingly confident as they rewrite, and the final product is indisputably their own. If I'd done this revision with the student, it would have been better in some ways. However, I would have altered her ideas in revising (I can imagine myself saying, "You have to say what kind of guy he is—a nice guy?"), thus reinforcing her notion that she cannot write, and I can. Using this program, she learns to go through the writing process on her own. Her third draft may not be a vast improvement over the first draft, but it's clearer. The crucial point is not the improvement, but the fact that this student is learning how it is done.

First draft: Michiel[1] is twenty five years old
He lives in California
Three weeks ago *he got an accident*[2]
He got therr brothers[3]
People thinks he is a guy[4]

Second draft: Michiel is twenty five years old
He lives in California
Three weeks ago he got an accident
He got therr brothers
People thinks he is a guy
He is a musician
Michiel Jacson loves his parents
All teenagers around the world loves him

Third draft:
[1]*Michiel Jacson* is twenty five years old
He lives in California.
He is a great musician for along time.
Michiel loves his parents and he has three brothers.[3]
People think he is a guy.[4]
Michiel has been in accident.
The accident is caused by fire.[2]
All teenagers around the world loves him

The first part of this program is a brainstorming session. The program gets the student thinking about the topic. Also, the student learns that writers ruminate before they write. The next part elicits a first draft. Some students use ideas from the brainstorming session, and others don't. The next part is the first revision; the program asks specific, meaningful questions on content encouraging the student to evaluate the clarity of what she's written. The program also foresees likely grammar problems and finds out what the student's grammar problem areas are by asking the results of a diagnostic program. It helps the students check for likely problems. The student learns to look at each sentence for typical errors when proofreading. Also, the program asks the student to judge organization. For example, do added ideas necessarily belong at the end? Would they make more sense at the beginning? The program can swap them around and let the student decide what order makes the most sense. The student learns to think about organizational revision, rather than presenting ideas in the order in which they occured.

Finally, the program has the student do another revision. By now students have often thought of improvements the program had not suggested, and they may add these, as well as more information. The program asks the student to continue writing, using the same revision techniques. By the end, the student has a composition which might be further revised, possibly on a word processor. More importantly, the student has learned something about the writing process.

I think these programs meet some of the criteria for an Artificially Intelligent Tutoring System proposed by Mead (1983). First, these programs are not limited by the need to input only one correct answer, and in fact respond meaningfully to a wide variety of correct and incorrect answers. Second, they can gain and use information from users either by asking questions or by making their own analysis of input. Third, the diagnostic, controlled composition, and proofreading exercises respond to student input in terms of the global meaning of the paragraph with which they deal, just as a human tutor would. Fourth, the free composition programs are capable of helping students evaluate their own writing in a way that simulates human reaction to student writing. It is not identical to human-student interaction, and in many ways is inferior, but in some ways, it may be superior because of the very nature of the way some students interact with computers.

Not all students respond to the programs in the same way. It has not been possible to compare students who use the programs with students who do not. Some students are told to use the programs, and others use them voluntarily. Other teachers at University of Massachusetts/Boston have had their entire class use programs, or have assigned certain students to use them. Certain teachers don't use them at all. Given these circumstances, it would be difficult to establish

comparable experimental and control groups, and difficult to quantify improvement in writing. Since the programs are supplemental, they do not do any harm. Almost all students enjoy the programs, and many seem to show a better understanding of the composition process and better control of mechanics after using them. Further, some students seem to do better with computer tutoring than with human tutoring for reasons that probably have to do with lowered anxiety.

The code used to write these programs is a very simple version of BASIC. Any version of BASIC would include the statements used: Print, Input, If-then-goto, Let and Remark. Although these programs run on a CYBER mainframe, I have written similar programs in Applesoft BASIC on an Apple IIE and Apple II PLUS, demonstrating that fairly sophisticated programs can be created using simple code.

There is no reason to be limited to the electronic flashcard approach which usually consists of multiple choice or fill in the blank drills in which the student types a single letter or word. BASIC handles *string-matching*, the comparison of groups of words such as sentences, very well. Imaginative courseware such as that described by Higgins and Johns (1984) can be created by taking advantage of BASIC's string-matching capabilities.

REFERENCES

Coburn, Peter, Peter Kelman, Nancy Roberts, Thomas Snyder, Daniel Watt and Cheryl Weiner. 1982. *Practical guide to computers in education*. Reading, Massachusetts: Addison-Wesley.

Higgins, John and Tim Johns. 1984. *Computers in language learning*. Reading, Massachusetts: Addison-Wesley.

Krashen, Stephen. 1982. Theory versus practice in language teaching. In *Innovative approaches to language teaching*. Robert Blair (Ed.). Rowley, Massachusetts: Newbury House.

Mead, William. 1983. A course in C.A.L.L. for an M.A. program in TESL. *TESOL Newsletter* 17(4):26.

Papert, Seymour. 1980. *Mindstorms: children, computers and powerful ideas*. New York: Basic Books.

Strunk, William and E.B. White. 1979. *Elements of style*. New York: MacMillan.

Thompson, Fred and Beth H. Thompson. 1975. Practical natural language processing: the REL system prototype. *Advances in Computers* 13:346.

Woods, W.A. 1978. Semantics and quantification in natural language question answering. *Advances in Computers* 17:3. New York: Academic Press, Inc.